**ITA 3: Problem Solving Methodology and Database software**

**Chapter 2, Developing a solution using database software, p 37 - 43**

In the previous chapter the problem solving methodology was introduced as a way of solving information technology problems.

There are seven stages of which only six are covered in this unit with documentation not covered.

The keyword used to describe the stages is: **ADDTDIE**

We have already dealt with in detail the analysis and design stage. In the design stage the last aspect of it we examined was the use of a Project Management approach to Problem Solving.

In this chapter we will through a database examine in detail at the problem solving methodology.

**Problem-Solving Methodology:**

This is a structured process to determine a solution to a problem. It involves the following seven steps:

**1. Analyse the problem:**

* The solution can’t be implemented until the problem has been properly defined and analysed.
* The problem is investigated.
* The problem is the defined.
* Alternative solutions are considered and a preferred solution is determined.

In the example of Ralph’s roofing, Ralph’s problems are that multiple tasks to a single customer are hard to keep track of. Also some of his payments have not been spotted and have fallen well behind. Also sometimes the writing is not neat and cannot be read by others.

**2. Design the Solution**

* Producing a plan that includes working out the hardware and software that is required to build the solution.
* It needs to be decided what data will be validated and stored and what procedures should be followed.
* A testing plan needs to be outlined to see what needs to be tested.

**Structure of databases:**

There are two distinct types of databases, Flat File Database and Relational Databases.

**Flat file database-** stores data in tables consisting of rows and columns.

**Field**- contains the same type of data for a series of records. It is the cateogory the information is entered into.

**Record-** is a set of information about one entity; for example a person, event or object.

**Primary key-** uniquely identifies each record in a data base table.

**Form or data entry screens-** allows an input screen to be formatted and linked to an underlying table or query.

**Query-** Records can be sorted and filtered out from the number of records by performing a query. Each query can be sorted using a number or a piece of text. This will eliminate the people or thing you are not looking for.

**Report-** formats the query data and allows you to add summary totals, as well as heading to make the information easier to understand.

**Macros or buttons or use of scripts-** are used to perform a certain set of tasks and they are run automatically.

**List the common data types-** Common data types are text, numeric (numbers only), data (a variation of numeric fields.), Boolean (Yes/No, True/False) or objects (images and video

In Filemaker pro the common data types are numerical and text.

**Relational databases**

Relational databases store data tables that are arranged in rows and columns. Tables are linked by a common field.

One-to-one relationships are used where a record in one table is connected to only one record in a second table. An example is an airline where the passengers details are stored, while a seat allocation table may hold records that relate to the seats on a flight. Each passenger has only one seat and each seat can only be assigned to one passenger.

One-to-many relationship indicates that one record in the first table can be connected to more than one record in the second table. For example several workers in a office may share a telephone extension. Each extension is related to several employees’ records.

Many-to-Many relationships indicate that one record in the first table can be connected to more than one record in the second table. At the same time each record in the second table can be connected to the first. An example is that students study many subjects and each subject has many students.

**Exercises on databases:**

**Complete the document, Databases, Flat File Vs Relational**